A comparison of 3D scan matching techniques for autonomous robot navigation in urban and agricultural environments

Javier Guevara[a], Jordi Gené-Mola[b], Eduard Gregorio[b], Miguel Torres-Torriti[c], Giulio Reina[d], Fernando A. Auat Cheein*[a]

[a]Department of Electronic Engineering, Universidad Técnica Federico Santa María, Valparaíso, Chile.
[b]Department of Agricultural and Forest Engineering, University of Lleida, Spain.
[c]Department of Electrical Engineering, Pontificia Universidad Catolica de Chile, Chile.
[d]Department of Mechanics, Mathematics and Management, Politecnico di Bari, Italy.

Abstract. Global navigation satellite system (GNSS) is the standard solution for solving the localization problem in outdoor environments, but its signal might be lost when driving in dense urban areas or in the presence of heavy vegetation or overhanging canopies. Hence, the need of alternative or complementary localization methods for autonomous driving. In the last years, exteroceptive sensors have gained much attention due to significant improvements in accuracy and cost-effectiveness, especially 3D range sensors. By registering two successive 3D scans —known as *scan matching*–, it is possible to estimate the pose of the vehicle. This work aims to provide in-depth analysis and comparison of the state-of-the-art 3D scan matching approaches as a solution to the localization problem of autonomous vehicles. Eight techniques (deterministic and probabilistic) are investigated: iterative closest point (with three different embodiments), normal distribution transform, Gaussian mixture model, support vector-parametrized Gaussian mixture and the particle filter implementation. They are demonstrated in long path trials in both urban and agricultural environments and compared in terms of accuracy and consistency. On one hand, most of the techniques can be successfully used in urban scenarios with the probabilistic approaches that show the best accuracy. On the other hand, agricultural settings have proved to be more challenging with significant errors even in short distance trials, due to the presence of featureless natural objects. The results and discussion of this work will guide the reader into selecting the most suitable method and encouraging to build improvements on the identified limitations.

Keywords: Autonomous vehicles, 3D point cloud registration, mobile robot sensing, robot localization.

***Fernando Auat Cheein,** fernando.auat@usm.cl

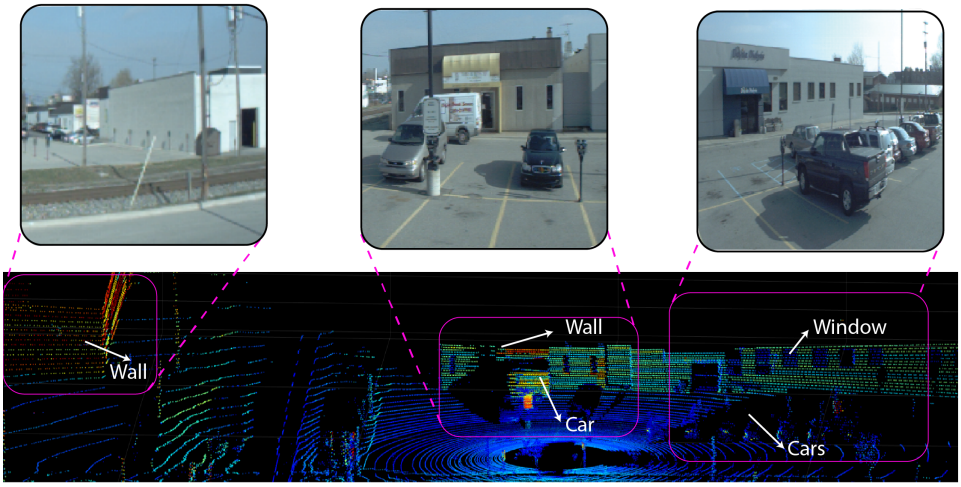
1 Introduction

Efficient and safe navigation of autonomous vehicles relies on the accuracy of their localization system.[1] A poor position estimation dramatically increases the probability of collision, compromising the integrity of the vehicle, the surroundings, and the task being performed. At present, the global navigation satellite system (GNSS) is the most widely adopted solution for localization. However, the position estimation accuracy depends on the available constellation and the quality of the signal.[2] In nominal conditions, the uncertainty could be of a few centimeters up to few meters.[3] In dense urban areas or agricultural environments with dense foliage, the GNSS signal might

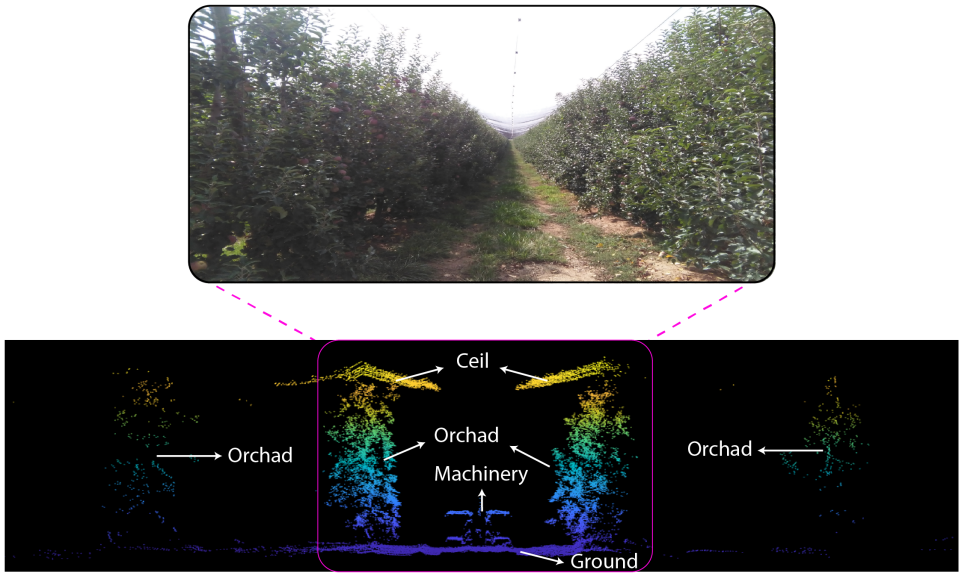
be simply lost, making unreliable the localization of the vehicle. More critically, in underground mining sites, there is a complete absence of GNSS signal. Additionally, the costs associated with GNSS antennas with centimeters (or even millimeters) of accuracy are usually considerably higher than standard portable receivers. Therefore, alternative solutions to the localization problem are necessary for replacing or complementing GNSS-based approaches.

To this aim, several simultaneous localization and mapping (SLAM) methods have been proposed in the last years (see[4,5] and the references therein). Nevertheless, the fact that the GNSS error is absolute, still represents an advantage when compared to SLAM based approaches or similar, where the error may accumulate over time or the accuracy depends on the loop closure (see[4]). For example, for dead-reckoning localization –particularly useful for short-path navigation–, the error comes from different sources (e.g., wheel slippage, misalignment, and terrain perturbations), causing its unbound growth over time.[6,7] Similarly, inertial navigation systems suffer from integration drift: small errors in the measurement of acceleration and angular velocity are integrated into progressively larger errors in velocity and position,[8] making inertial navigation challenging to use in the long path.[?,9]

Range sensors, such as light detection and ranging (LiDAR) has brought attention to autonomous vehicles development, mainly for their decreasing costs and high accuracy.[10–12] The LiDAR-based point cloud information can be used to accomplish localization purposes when using scan matching techniques.[13] These techniques aim to estimate the rigid motion transformation, which maximizes the overlap between two frames obtained at different time. Scan matching techniques have been widely used as a localization system either with 2D (3 DoF)[14,15] or 3D laser scanners (6 DoF),[16,17] showing suitable solutions for indoor environments[12] when assuming objects of only polygonal shapes.[15] However, when applying scan matching to unstructured environ-



(a)



(b)

Fig 1: Raw point cloud data and their corresponding images for (a) urban and (b) agricultural environments.

ments, the performance of the registration techniques degrades.[18] For example, a comparison of scan matching techniques in real-world data sets showed the limitations of several scan matching methods when applied to unstructured environments.[19] To improve registration in a specific environment, works such as[20,21] proposed the extraction of specific patterns (normals from the point

cloud, or reflectance information from the LiDAR readings). However, these variations are not expected to work in all environments since normals are not always available or interpretable, and reflectance varies significantly for each environment. Figure 1 shows the two cases tackled in this work: an urban and an agricultural setting, with their corresponding point clouds obtained using a LiDAR. The urban scene has distinguishable planar surfaces, such as walls, windows and cars. In contrast, the agricultural setting has irregular point clouds on the trees of the orchard.

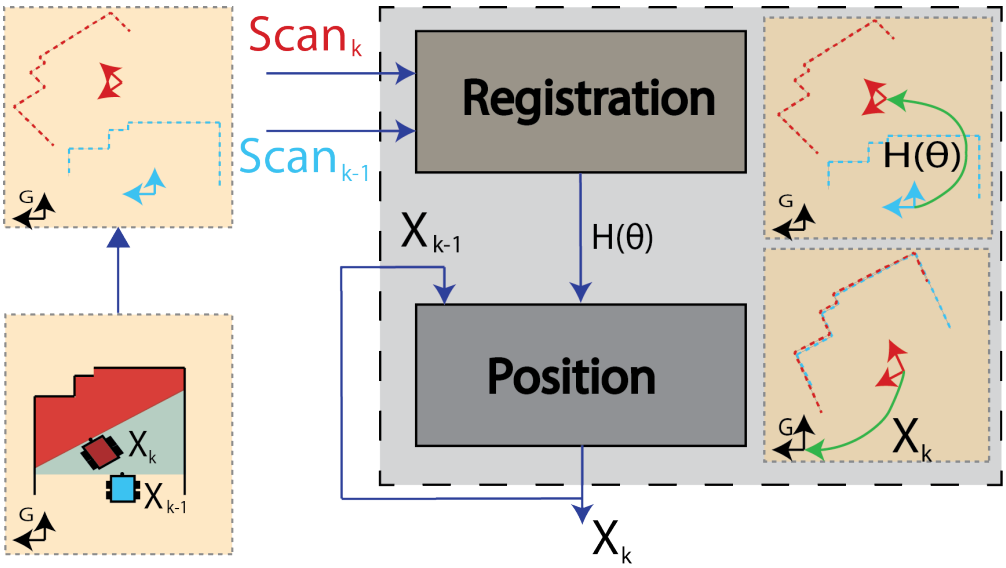


Fig 2: The scan matching takes as input two scans obtained at different time, and they are aligned using a registration algorithm with output $H(\theta)$ which represents the rigid transformation matrix associated with the sensor displacement. The later is then associated with the vehicle motion.[13] The position, X_k –suffix k stands for sampling time– of the sensor (and therefore, of the vehicle) is obtained concatenating the transformation matrices causing the error in localization to be cumulative.

Figure 2 depicts the scan matching process for vehicle localization, assuming the sensor is mounted on the vehicle (which is the case in this work). The key limitation of applying scan matching to estimate the pose of the vehicle lies in the dependence of the previous state. Once a position and orientation are computed, the vehicle does not revisit the place, and it cannot recover from possible registration error.[7] Typically, a loop closure approach is used to mitigate this error in SLAM algorithms. In this work, we consider the scan matching techniques as a self-positioning

system without the need to close the loop; therefore, cumulative errors are expected.

Registration algorithms consider deterministic and probabilistic data association approaches. The former is a particular case of probabilistic methods, where uncertainty is zero. For example, the Iterative Closest Point (ICP)[17] and the Polar Scan Matching (PSM)[22] find correspondences among points using the Euclidean distance in Cartesian and polar coordinates, respectively. Similarly, the Iterative Dual Correspondence (IDC)[23] incorporates a matching range point to define correspondences and to improve data association.

On the other hand, the probabilistic approaches take into account the uncertainties in the sensor measurements to implement a maximum likelihood estimator (MLE). However, when using a probabilistic approach with the wrong parameters, the uncertainty can produce worst results than deterministic approaches. Some examples of these approaches are the generalized-ICP (GICP),[24] and the normal distribution transform (NDT).[25] The latter describes the point cloud by a set of local probability density functions using a voxel-based structure. This, however, is one of the main disadvantages since there is not a validated method of selecting the right voxel size. When the cell is significantly big, the computational time is low, and the accuracy decreases. In contrast, when the voxel size is small, the accuracy increases, but it comes with a high computational cost. Variations of NDT include a pre-processing stage,[26,27] where the objects in the scene are grouped according to their similarities. For example, in,[26] it is used edges and planes acquired from the scene to differentiate objects. Further, in[27] the differentiation is improved by incorporating the POINTNET++ network,[28] which is used for applying semantic segmentation in the scene. After the label assignment, the NDT is performed individually to the same label objects, and the rigid transformation is obtained by minimizing the sum of all rigid transformations.

Regarding probabilistic approaches, a Gaussian mixture models (GMM) representation could

also be used for performing registration, as shown in.[29] The GMM approach can deal with noise and outliers to some extent,[30] unlike the previously mentioned approaches (NDT, GICP). Different variations of GMM registration algorithms rely on the need for a pre-processing stage, which is the case of the Support Vector-parametrised Gaussian Mixture (SVR).[31] Other probabilistic registration algorithms are based on filtering theory: they can use prior information in a maximum posterior (MAP) sense.[32] The registration methods that use MAP estimator often use Bayesian filters, such as Kalman or Bingham filter.[33] A significant disadvantage of these filtering-based methods is the requirement of tuning several parameters, which can be counter-intuitive.

This work aims to lead the readers into the different registration algorithms existing in the literature and show how suitable they are for addressing the positioning problem for reliable autonomous navigation in two specific and yet different environments: urban and agricultural, assuming exteroceptive sensors mounted on the vehicles.

Although several scan matching techniques have been proposed during the past decade [28], we try to cover some of the most representative deterministic and probabilistic approaches in this work. To this aim, we have selected a variety of open-source approaches (see Appendix), which have been widely used by the scientific community because of their easily obtainable implementation. We begin by describing the well-known ICP and its variations to point-to-point and point-to-plane. Then, we describe the GICP, the NDT, the GMM, the coherent point drift (CPD), the SVR, and the Particle Filter registration (PF) algorithm. We analyze the performance of the previously mentioned techniques under different navigation trials. The comparison of all the algorithms is made with real data from urban and agricultural environments under real field conditions, acquired by two vehicles (a car and an autonomous agricultural platform, respectively), with a 3D LiDAR sensor, thus offering an in-depth insight of such techniques with long path field results.

2 Mathematical background

Let S and T be a source point cloud and a target point cloud, respectively. The goal is to find the rigid transformation matrix that aligns S with T as shown in Fig. 3. For this purpose, a transformation matrix, H, with parameters $\theta = [x, y, z, roll, pitch, yaw]^T$ is applied to the source point cloud, as $H(S, \theta)$. Such transformation is obtained through an iterative procedure, which consists of a maximum number of iterations, I_{max}, and an error threshold, ϵ.

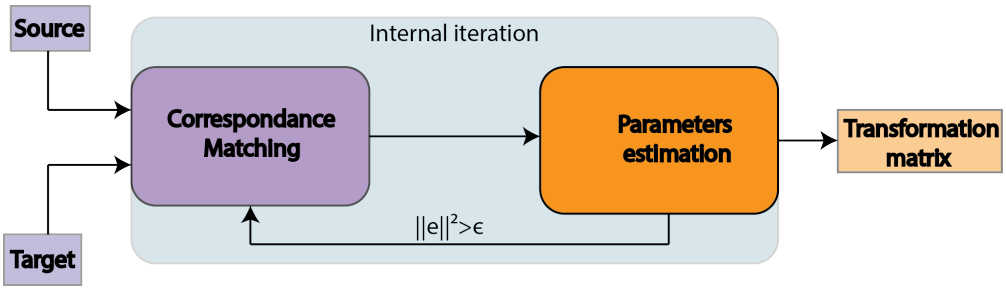


Fig 3: Procedure for point cloud registration. The inputs are two point clouds, a source and a target, and the aim is to find a correspondence among them and, by minimizing their discrepancy. The process is repeated until the per-residual error among points, e, is less than ϵ or a maximum number of iterations, I_{max}, has been reached.

Figure 3 depicts the registration procedure to obtain the transformation matrix, H, with rotation R and translation t_r. The procedure takes as input two point clouds: a source, $S = \{s_i\}_{i=1,...,m}$; $S \in \mathbb{R}^3$, and a target, $T = \{t_i\}_{j=1,...,n}$; $R \in \mathbb{R}^3$. Herein, we consider M and N as the number of points in the source and target point cloud, respectively. The first step is to find the correspondence between the two point clouds and then, an optimization procedure finds the registration parameters θ. Then, the procedure iterates until the registration error e is less than ϵ or I_{max} is reached. This applies to all scan matching techniques analysed here.

2.1 The iterative cloud point (ICP) and its variants

The ICP considers the raw data from both point clouds. Different metrics can be used as a distance between the point sets, e.g., point-to-point and point to plane.[19,34,35] The point-to-point method minimizes the sum of squared distances between each corresponding pair of points. On the other hand, the point-to-plane method minimizes the distance between the points and the tangential planes at the corresponding nearest points. The correspondences between $H(S,\theta)$ and T are computed using the nearest neighbour criteria $Y_j = ClosesPoint(H(S,\theta),T)$. For computing the closest point, the metric used is the Euclidean distance. Further, the transformation matrix is obtained with the parameters that minimize the L_2 error, as shown in Eq. 1.

$$\theta \leftarrow \underset{\theta}{argmin} \sum_{j=1}^{n} \eta_i \|Y_j - H(S,\theta)\|^2 \tag{1}$$

where η_i is the surface normal at m_i and it is used in point-to-plane registration. The minimization of Eq. 1 is done using a least-squares approach.[17] When the set of points are far away among each other, the nearest neighbour point does not correspond, in general, to the same point on the target point cloud, especially when using the Euclidean distance.[36] Other approaches use different distance criteria; for example, the Mahalanobis distance[32,37] and the most likely criteria;[32,38] or assigning a weight to each correspondence alternatives.[39,40] Therefore, the points further apart have lower weights than points with close neighbours. However, it is still not possible to get precise correspondences even after reaching convergence.[41]

In,[24] it is presented a generalization of the ICP, named Generalized-ICP (GICP). The GICP incorporates a probabilistic framework where the covariance matrices are associated with each point from both point clouds. Such approach considers that $s_i \sim \mathcal{N}(\mu_i^s, \Sigma_i^s)$ and $t_i \sim \mathcal{N}(\mu_i^t, \Sigma_i^t)$

are drawn from independent Gaussian distributions and the correspondences are also computed using Euclidean distance. For an arbitrary rigid registration H, $d_i^{(H)} = t_i - H(s_i, \theta)$. If H^* is the correct transformation, then, $t_i = H(s_i, \theta)$. Therefore, $d_i^{H^*} = \mathcal{N}(0, \Sigma_i^t + (H^*)\Sigma_i^s(H^*)^T)$. When using the maximum likelihood estimator to iteratively compute H, the parameters are obtained as shown below,

$$\theta = \underset{\theta}{argmax} \prod p(d_i^H) = \underset{\theta}{argmax} \sum_i log(p(d_i^H)).$$

The latter can be simplified to:

$$\theta = \underset{\theta}{argmin} \sum_i d_i^H (\Sigma_i^T + H\Sigma_i^S H)^{-1} d_i^H$$

2.2 The coherent point drift (CPD)

The CPD is highly suitable for accurate point cloud registration.[42] However, its computation complexity is extremely high, which is a problem in large-scale point clouds. The CPD describes the registration as a Gaussian mixture models problem, where S considers the GMM centroids and T the data points generated by the GMM. Therefore, the probability density function is described as $p(t) = w\frac{1}{N} + (1-w)\sum_{m=1}^{M} \frac{1}{M} p(t|m)$, where $p(t|m) \sim \mathcal{N}(\mu_m^s, \Sigma)$ and w is the weight of the uniform distribution. This approach uses expectation maximization to find the parameters of the rigid transformation. The expectation can be seen as the correspondence matching, which is based on the posterior probability of the GMM centroid given the data, described in Eq. 2.

$$P^{old}(m|t_n) = \frac{e^{-0.5||\frac{t_n - H(s_m,\theta)}{\sigma^{old}}||^2}}{\sum_{k=1}^{M} e^{-0.5||\frac{t_n - H(s_m,\theta)}{\sigma^{old}}||^2}} \tag{2}$$

The new parameters' values are found by minimizing the expectation of the complete negative log-likelihood function, as shown in Eq. 3 below,

$$Q = \frac{1}{2\sigma^2}\sum_{n=1}^{N}\sum_{m=1}^{M} P^{old}(m|t_n)||t_n - Rs_m - t_r||^2 + \frac{N_p}{2}log(\sigma^2) \tag{3}$$

where $N_p = \sum_{n=1}^{N}\sum_{m=1}^{M} P^{old}(m|t_n)$. Considering $P = P^{old}(m|t_n)$, the point clouds are centred: $\hat{T} = T - T^T P^T/N_p$ and $\hat{S} = S - S^T P/N_p$. By applying the singular value decomposition (S, V, U) of $\hat{T}P^T\hat{S}$, where $UU^T = VV^T = I$ and $SS = d(s_i)$ with $s_1 \geq s2 \geq, ..., \geq s_D \geq 0$, the parameters are updated as follows:

$$R = UCV^T;\ t = \frac{1}{N_p}T^T P^T - R\frac{1}{N_p}y^T P;\ \sigma^2 = \frac{1}{N_p}\left(tr(\hat{X}^T d(P^T)\hat{X}) - tr(\left(\hat{T}P^T\hat{S}\right)^T R)\right)$$

where $C = d(1, ..., 1, det(UV^T))$.

2.3 The normal distribution transform (NDT)

The NDT first divides T into voxels and assigns a normal distribution transform to each one of them with mean u_i and covariance Σ_i. The goal is to find the transformation parameters that maximizes the likelihood, p_i, of points from S that lie on T. For this purpose, the correspondence between points in S and their voxels from T is obtained according to:

$$\tilde{p}(p_i) = -d_1\ exp\left(-\frac{d_2}{2}(p_i - \mu_i)^T\Sigma_i^{-1}(p_i - \mu_i)^T\right)$$

where $d_2 = -2log\left((-log\left(c_1 exp(-1/2) + c2\right) - d_3\right)/d_1)$,$d_3 = -log(c_2)$, $d_1 = -log(c_1 + c_2) - d_3$, and c_1 and c_2 are constant values related to the size of the cell. The NDT score function is defined as $s(\theta) = -\sum \tilde{p}\left(H(S,\theta)\right)$. Here, Newton optimization is used to optimize $s(\theta)$ and it is computed to solve the equation $\mathbf{H}\Delta\theta$ = -g, where $\mathbf{H}$ is the Hessian matrix and the g the gradient vector of the score. It is worth to mention that the NDT requires of high computing power capability, and the performance is directly related to the size of the cell.[41]

2.4 The Gaussian mixture model (GMM) registration

The GMM (see[43]) considers that the probability density function of a general Gaussian mixture is defined as $p(x) = \sum_{i=1}^{k} w_i \mathcal{N}(\mu_i, \Sigma_i)$, where: w_i is a the Gaussian weight. The input point cloud is represented by $GMM_S = \sum_{i=1,\ldots,m} w_i \mathcal{N}(\mu_i, \Sigma_i)$ and $GMM_T = \sum_{j=1,\ldots,n} w_j \mathcal{N}(\mu_j, \Sigma_j)$. As can be seen, the number of Gaussian components is the number of points in the cloud, and all components are equally weighted. In addition, for each component, the mean vector is given by the spatial location of each point and all components share the same covariance. An optimization problem can be then proposed to find the transformation $H(\theta)$, minimizing the L_2 distance of the two GMM as follows:

$$\theta \leftarrow \underset{\theta}{argmin} L_2\left(GMM_T, GMM_{H(S,\theta)}\right). \tag{4}$$

Using the close-form expression of L_2[44] with GMM, Eq. 4 becomes:

$$\theta \leftarrow \underset{\theta}{argmin} \sum_{(i,j)\in\mathcal{R}} w_i w_j \mathcal{N}(0|t_i - H(S,\theta), \Sigma_T + \Sigma_{H(S,\theta)}). \tag{5}$$

Such optimization could warranty only a local minimum, and the results are related to the Gaussian smoothness. Therefore, the parameter σ can be decreased in order to expand the search area. In,[31] it is presented a variation of GMM registration (SVR), which includes a pre-stage to the GMM registration, as a

one class support vector machine with a Gaussian radial basis function kernel, as shown in Eq. 6,

$$f(x) = \sum_{i=i}^{l} \alpha_i e^{-\gamma||x_i - x||^2} - \rho \tag{6}$$

where γ is the Gaussian kernel width, x_i is the point vector, α_i is the weight, x is the input vector, ρ is the bias and l is the number of training samples. The output of the support vector machine involves a sparse subset of the data points, which is later used to perform registration with a GMM representation.

2.5 The particle filter registration (PF)

The PF, for scan registration purposes, is adapted to perform a variation of the iterative closest point.[45] The registration procedure considers a state-space model $x(k) = \theta(k)$, and the observation space given by $z(k) = \theta^m(k)$. First, N particles $\{x^i; i = 1, ..., N\}$ are initialized, and the transformed point set is selected as the measurement. Then, it iteratively proceeds as follows: first, a motion error for each time k based on the predicted and measured state at the previous time is computed as shown in Eq. 7 below,

$$e\left(x^i_{k-1}, \hat{x}^i_{k-1}\right) = x^i_{k-1} - \hat{x}^i_{k-1}. \tag{7}$$

Second, N_p particles are drawn based on the proposal density presented in Eq. 8,

$$q(x_k|x^i_{k-1}, z_k) = \frac{1}{N_p} K \left(\frac{x^i_k - x^i_{k-1}}{\Sigma^i_{k-1}} \right) \tag{8}$$

where Σ^i_{k-1} is the covariance motion error, and K is a Gaussian function. The next step is to minimize $E = \sum_{j=1}^{n} ||T_j - RS_j - t_r||^2$ for L iterations, which is the same objective function presented for the ICP case in Section 2.1. Weights are updated according to $w^i_k = w^i_{k-1} p(z_k|\hat{x}_k)$.[46] Once the weights are updated, the cumulative distribution function is built, and the particles are re-sampled.[46] Finally, the parameters that minimize the objective function are selected. As can be seen, particle-filtering involves additional steps

to the basic ICP. A significant disadvantage of this method is the tuning of several parameters such as the number of particles, N_p, the number of iterations L, and those associated with the Gaussian function, K.

3 Experimental data sets

For the urban scan matching evaluation, we considered the publicly available Ford Dataset,[47] which was generated using a Ford F-250 vehicle (© Ford Motor Company, S.A., Dearbon, Michigan, USA) equipped with Velodyne HDL-64E laser scanner (Velodyne LIDAR Inc., San José, CA, USA), and Applanix POS-LV 420 INS with Trimble GPS (Timble Inc., Sunnyvale, California, USA) used for ground truth data. The LiDAR was mounted horizontally and the data was captured with the laser spinning at 10 Hz. A single LiDAR raw data contain approximately 80.000 points (acquisition speed of 800.000 points/ second).

For the scan matching assessment in agricultural environments, we generated our own data set. This data was acquired in a commercial Fuji apple orchard (Malus domestica Borkh. cv. Fuji) located in Mollerussa, Catalonia, Spain (41°36'48.5"N, 0°51'41.7"E). Trees grown in the selected orchard were trained in a tall spindle system with a maximum canopy height of 3.5-4 meters, width of 1-1.5 meters, and tree spacing of 4 x 1 meters. Data was acquired on July 29th of 2019, when trees were at BBCH (Biologische Bundesanstalt, Bundessortenamt und Chemische Industrie) growth stage 75,[48] with fruits about half final size.

The autonomous platform depicted in Fig. 4 was used to acquire the data in the agricultural field. The platform consisted of an aluminum structure mounted on a continuous track composed of two rubber belts moved by two electrical ac motors. These motors were powered by a petrol-engine generator and controlled by two variable frequency drives, which were used to control the speed and the direction of advance. The platform was equipped with a LiDAR sensor and a GNSS-RTK mounted on a vertical mast of 2 meters height that was firmly fixed to the front of the platform.

The LiDAR sensor was a Puck VLP-16 (Velodyne LIDAR Inc., San Jose, CA, USA), which provides a three-dimensional point cloud per scan by means of 16 laser beams (905 nm wavelength), with a maxi-

mum range of 100 meters and an accuracy of $\pm$ 0.03 meters. This sensor was mounted in vertical position at a height of 1.8 meters, which corresponds to the half maximum height of studied trees. Mounting the LiDAR sensor vertically is a common practice for geometric characterization of vegetation to have a higher vertical resolution[49,50] . The scanning frequency rate was set to 10 Hz, corresponding to a vertical angular resolution of 0.2°. At this configuration, the LiDAR sensor acquired a maximum of 28,800 points (16 laser beams * 360° FoV / 0.2° resolution) per scan, corresponding to an acquisition speed of 288,000 points/second. The acquisition of Coordinated Universal Time (UTC) of each point was obtained via a GPS 18x LVC receiver (Garmin International Inc., Olathe, KS, USA), connected to the VLP-16 sensor. This GPS receiver was only used to obtain the UTC of each LiDAR point.The RTK-GNSS system was the GPS1200+ (Leica Geosystems AG, Heerbrugg, Switzerland), which obtains absolute coordinates at a frequency rate of 20 Hz with an error of 0.01/0.02 meters (horizontal/vertical). The GNSS rover antenna was mounted on the top position, at a height of 2 meters. Each sensor was connected to a rugged laptop GETAC V110 (Getac Technology Corporation, Baoshan, Taiwan) with a 64-bit operating system, 8 GB of RAM and an Intel Core i7-7600 U 2.70 GHz processor. LiDAR data was acquired using VeloView 3.5 software (Velodyne LIDAR Inc., San Jose, CA, USA), while GNSS data was acquired using a self-developed LabVIEW (National Instruments, Austin, USA) program which stores the receiver coordinates and the UTC time (synchronized with the LiDAR) of each positioning measurement. The scanning was performed driving the platform at a constant velocity of 0.5 ms^{-1} throughout five consecutive orchard alleyways (dirt road soil) of 250 meters long. The generated dataset (AgLiMatch dataset) has been made publicly available at http://www.grap.udl.cat/en/publications/datasets.html.

4 Experimental results

To assess the different scan matching techniques, a frame-to-frame registration on both datasets was performed, considering the raw point cloud data acquired by the LiDAR sensor. The metrics that we consid-

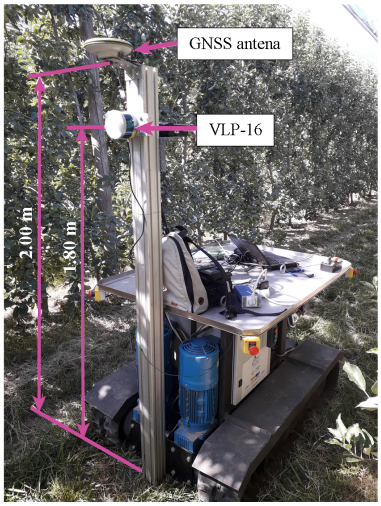


Fig 4: Autonomous platform used for agricultural environment scanning.

ered were the total root mean square error, $RMSE$, the translational, e_t, and rotational, e_r, errors against the ground truth. The $RMSE$ describes the total root mean square error per point j after registration as shown below,

$$RMSE = \sqrt{\frac{1}{N}\sum_{j=1}^{n} ||T_j - RS_j - t_r||^2} \tag{9}$$

where T, S, R and t_r denote the source point cloud, the target point cloud, the rotation and translation transformations, respectively. As we are mainly interested in both the translation and rotation, we projected the 6D distribution into the translation and rotation errors.[19] Considering the ground-truth transformation matrix, H_g, and the corresponding registration solution, H, we can define the remaining error ΔT as follows:

$$\Delta T = \begin{bmatrix} \Delta R & \Delta t \\ 0 & 1 \end{bmatrix} = H H_g^{-1} \tag{10}$$

with its translation error, et, defined as the Euclidean norm of translation vector ΔT:

$$e_t = ||\Delta t|| = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \tag{11}$$

and its rotation error, e_r, defined as the Geodesic distance directly from the rotation matrix ΔR:

$$e_r = arccos\left(\frac{trace(\Delta R) - 1}{2}\right) \tag{12}$$

4.1 Parameter selection

As described in Section 2, several parameters should be considered for the registration approaches. Finding an appropriate combination of parameters can vary according to each application.[51–56] Some parameters were fixed in this work and others were selected by analyzing the trade-off between speed and accuracy. Parameters such as the maximum number of iterations and the error threshold were set to 100 and 0.001, respectively. The parameters analyzed for each algorithm are described above. For the ICP in its point-to-point and point-to-plane version, we analyzed the use of a different percentage of paired points with the Euclidean distance, which is a common practice to robustness the registration procedure.[57] In the point-to-plane case, we use 10 points to the computation of the normals. For the GICP, we analyzed the use of the number of points to compute the covariance matrix. For the NDT, we evaluated different voxel sizes. For the CPD, we analyzed different weights, w, of the uniform distribution. For the PF implementation, we considered the use of a different number of particles. For the GMM and SVR, a much more involved process is needed to tune the whole set of parameters; therefore, we consider the parameters assigned in their original implementation. For evaluating the parameters mentioned above, we considered two sub-sampled[58] frames of an urban scenario with a known transformation matrix. Figure 5 depicts the algorithms speed against their translation, rotation, and the root mean square error. Regarding the ICP point-to-plane version, we selected 90 $\%$ of paired points, since the translation, rotation, and RMSE error decrease when the percentage of paired points increases. Similar behavior was obtained with the ICP point-to-point version, but the lowest error and time was obtained with 95 $\%$ of paired points. For the GICP, it is notorious that the average time to

convergence and the errors increase when more points are considered in the covariance matrix; therefore, we have selected twenty closest points to construct the covariance matrix. For the NDT, we prioritize accuracy over time; as it can be seen, the minimum error was obtained with a voxel of one meter. For the CPD, we have selected a weight for uniform distribution equal to $w = 0.5$ because a minimum variation was appreciated in all the evaluated weights. For the PF, it can be seen that ninety particles can obtain the lowest error and computational time.

4.2 Urban Dataset

Figure 6 shows the consistency analysis for the overall pose estimation and frame-to-frame registration. For a better understanding of the estimation results, we specified the trajectory scan number of five paths, named *A, B, C, D* and *E*.

To evaluate the pose estimation, we followed the guidelines presented in[59] to perform consistency tests. Figure 6 shows the consistency for the x and y coordinates of the complete experimentation. The results show that the ICP point-to-point, the ICP point-to-plane, the GICP, the NDT, and the CPD do not exceed in more than 5 $\%$ the maximum of twice their standard deviation, which suggests that such approaches could be used for fairly long distances. Nevertheless, the GMM, SVR, and PF registration show inconsistency in several parts of the road. The SVR, for example, shows consistency when the experiments start. However, close to the turn to take the A path, it becomes inconsistent. On the other hand, the GMM and the PF registration do not show consistency at all.

Figure 7 depicts a qualitative representation of the pose estimation for the Ford Dataset. Each algorithm shows the estimated path with the XY map projection of LiDAR data and the truth path obtained with the GNSS readings. The results showed that accurate positioning could be obtained with some registration algorithms, such as GICP and NDT. Regarding the GMM, the results were not satisfactory due to the lack of motion detected, consistently with the experiments developed by.[29] On the other hand, the SVR lost its

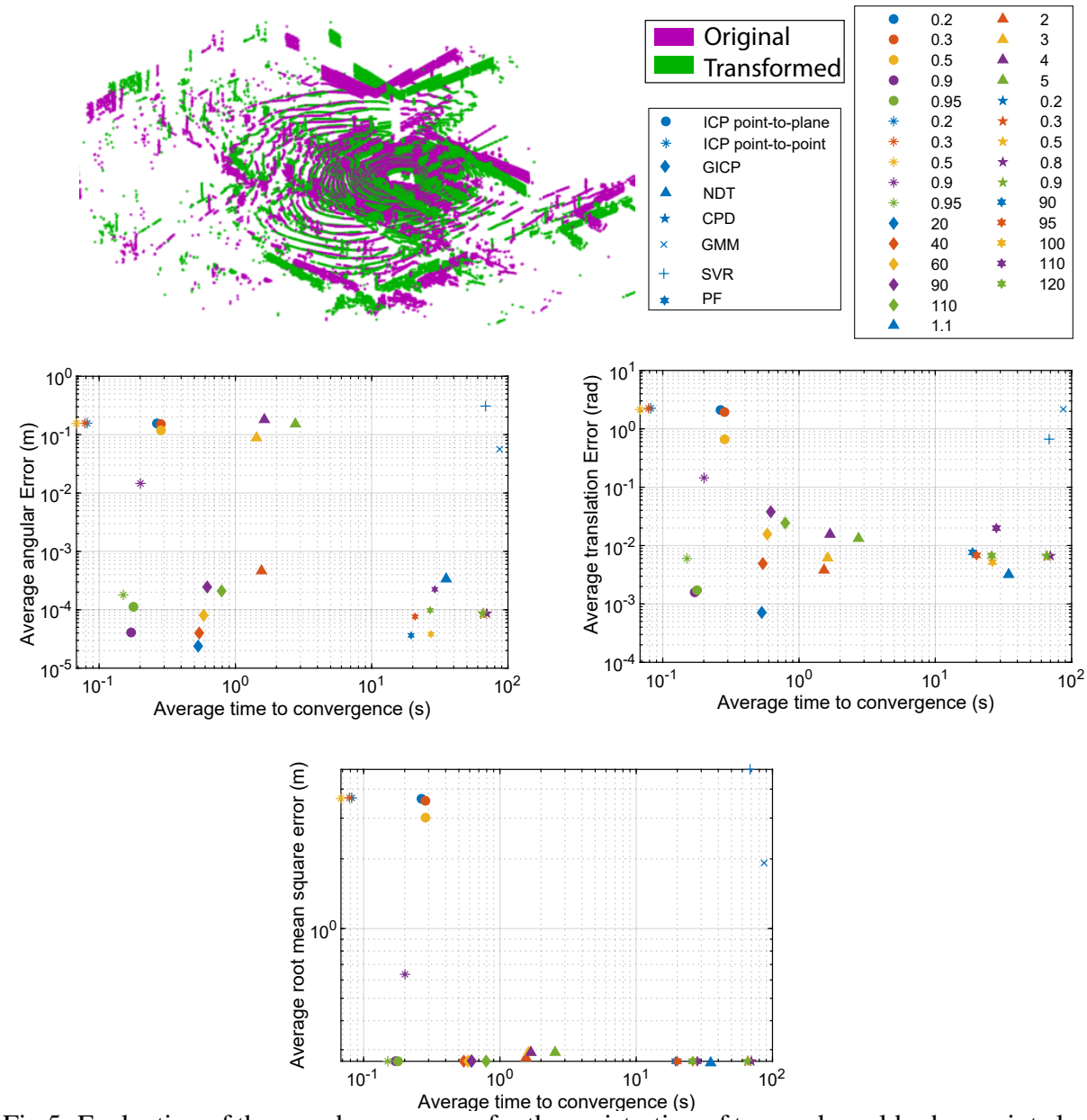


Fig 5: Evaluation of the speed vs accuracy for the registration of two real-world urban point clouds. The accuracy is evaluated considering the average translation error, the average rotation error and the average root mean square error after registration.

direction in the second turn, alike the PF at the beginning of the experiments.

4.3 Agricultural dataset

To evaluate the performance of the scan matching techniques in an agricultural scenario, five different tracks from the agricultural dataset were considered: two short path distance, two medium-path distance, and one

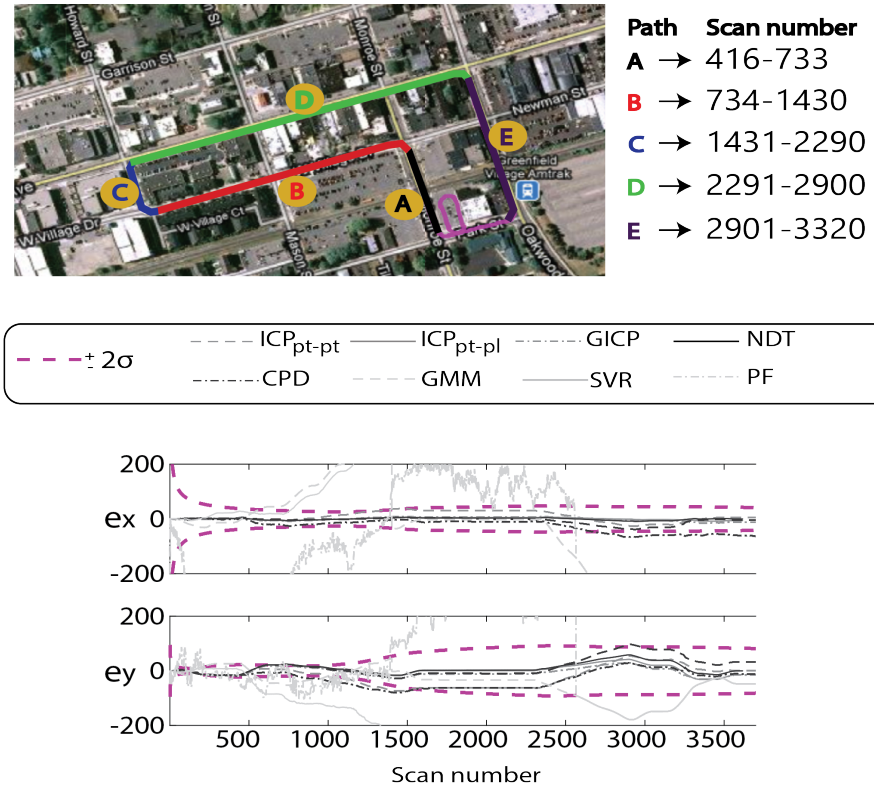


Fig 6: Results for Ford Dataset. On top, the path followed by the vehicle. The experimentation consists of 3800 frames of laser scanner. On the bottom, the consistency analysis for the vehicle position and the frame-to-frame registration; e_x and e_y represent the error in x and y coordinates of the estimated position, respectively.

long path distance. The path followed by the vehicle can be seen in Fig. 8, where $Sp_{k=i,\dots,5}$ denotes the starting point of each experiment.

In contrast to the urban scenario, the performance of the scan matching techniques degrades significantly with distance and manoeuvres. As it can be observed in Fig. 8, although tested trajectories were straight lines, all predicted trajectories –using different scan matching techniques– present significant misalignments in scale and rotation. The poor performance on this estimation was mainly due to the penetration of LiDAR beams into the vegetation, which produces a high-entropy point cloud with information from leaves, fruits, branches, ground, among others. Additionally, the scanned scene is non-static due to the movement of leaves under windy/outdoor conditions.

Figures 9 and 10 show a consistency analysis for position estimation and the frame-to-frame registration

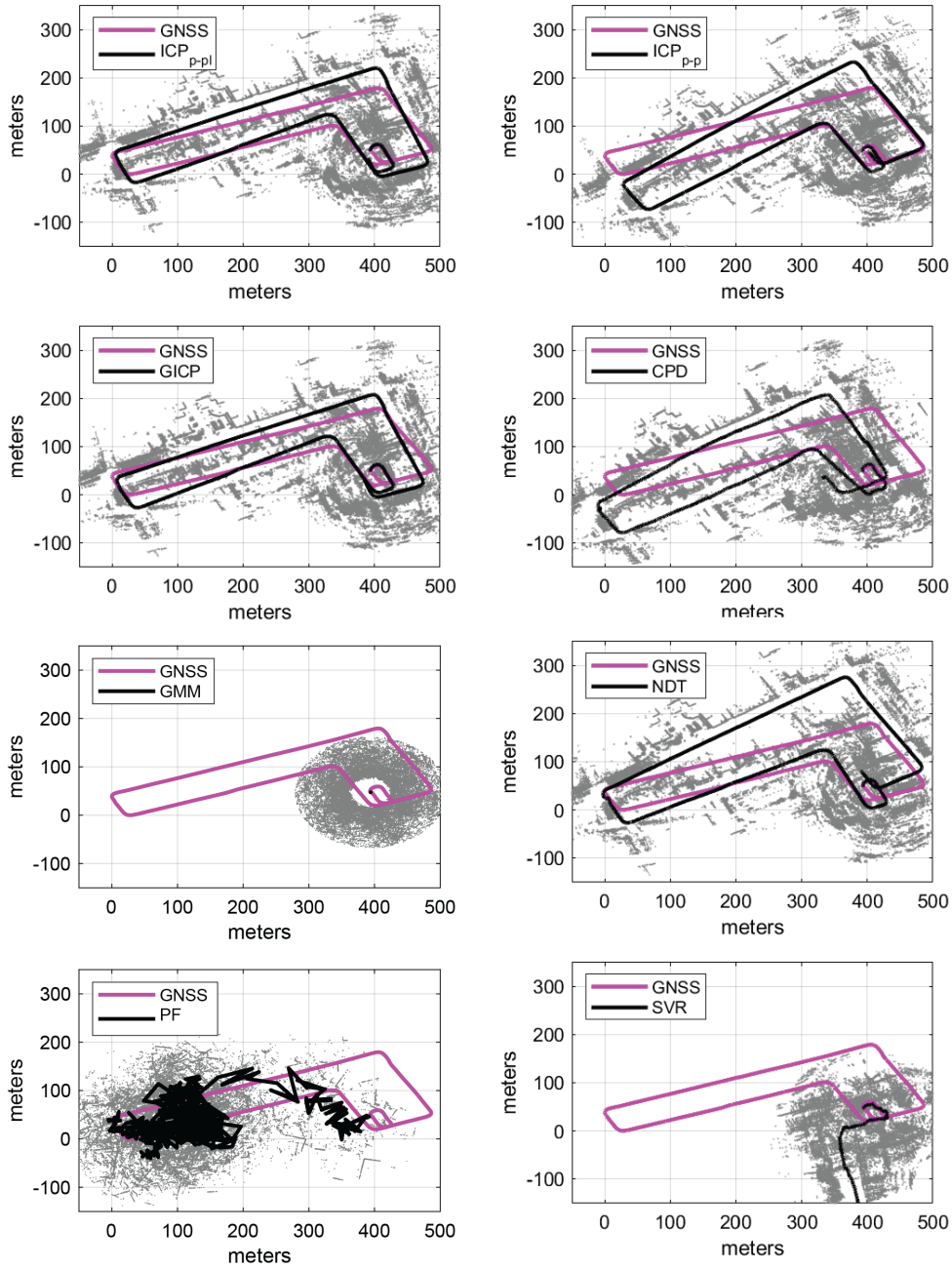


Fig 7: Paths reconstruction. Ground truth path shown in solid magenta line and estimated path shown in solid dark line. In gray, it is shown the 2D projection of the reconstructed map.

for each track experimentation. For short-path experimentation, it can be seen that the error of most of the approaches keeps under the standard deviation, however, the error reaches values of 50 meters in each axis, and the standard deviation presents a continuous growth. When analyzing the long path experimentation, similar behavior is presented, but the error increases up to 100 meters. Finally, the frame-to-frame consis-

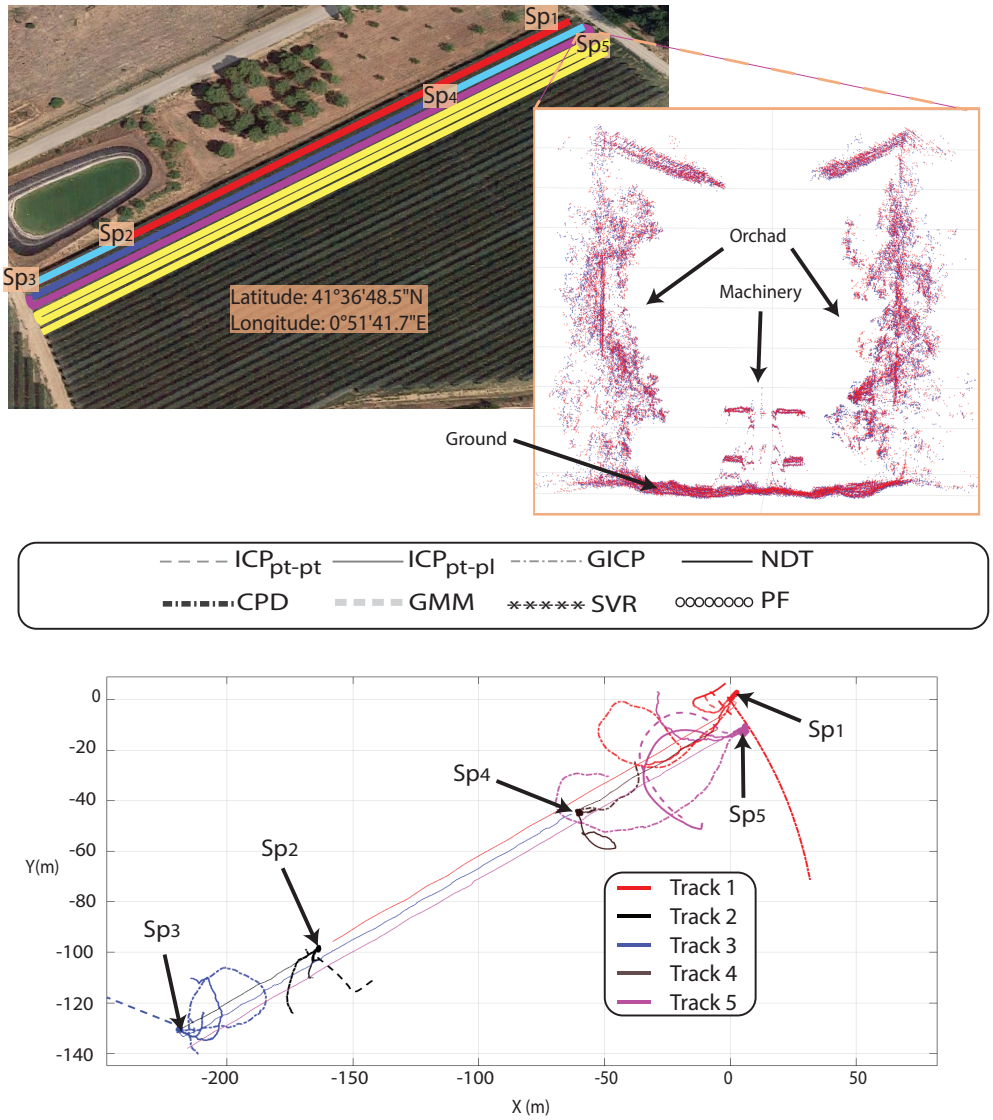


Fig 8: Results from the agricultural environment. On top, the paths followed by the vehicle; S_p denotes the starting point of each path. There are two short path trials, starting at S_{p2} and S_{p4}. The long path experimentation starts at S_{p5}. The medium path trials are labelled in red and blue, respectively, while the short paths are depicted in cyan; the long path is shown in purple. In yellow the rest of the point cloud obtained during trials. On the bottom, it is shown the estimated path according to each registration technique.

tency analysis shows that the translation become inconsistent in different parts of the followed path. It has to be noted that the error increase unbounded.

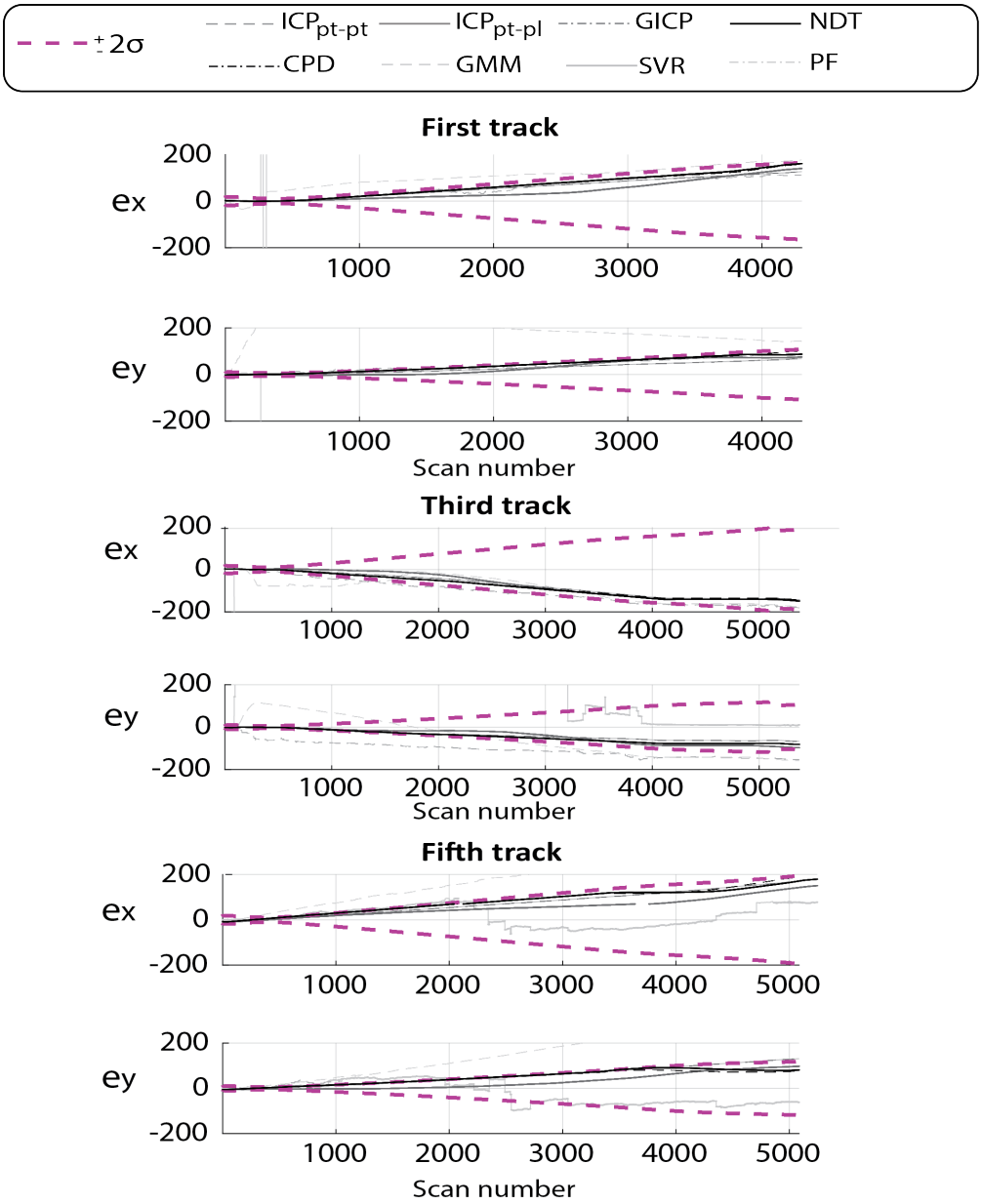


Fig 9: Consistency analysis of the vehicle position and the frame-to-frame registration for the medium path experimentation –first and third tracks–, and for the long path experimentation –fifth track. The left figure shows the positioning errors e_x and e_y according to x and y coordinates.

4.4 Evaluation of results

For evaluating the error distribution in both scenarios, we followed the guidelines described in.[19] To do so, we analyzed the median and the quantiles of the recall-accuracy threshold plots,[19] which compare the cumulative probability of translation and rotational error against the error magnitude. The quantiles are defined as A50 (i.e., the median), A75, and A95 and correspond to the probability of 0.5, 0.75, and 0.95 of the

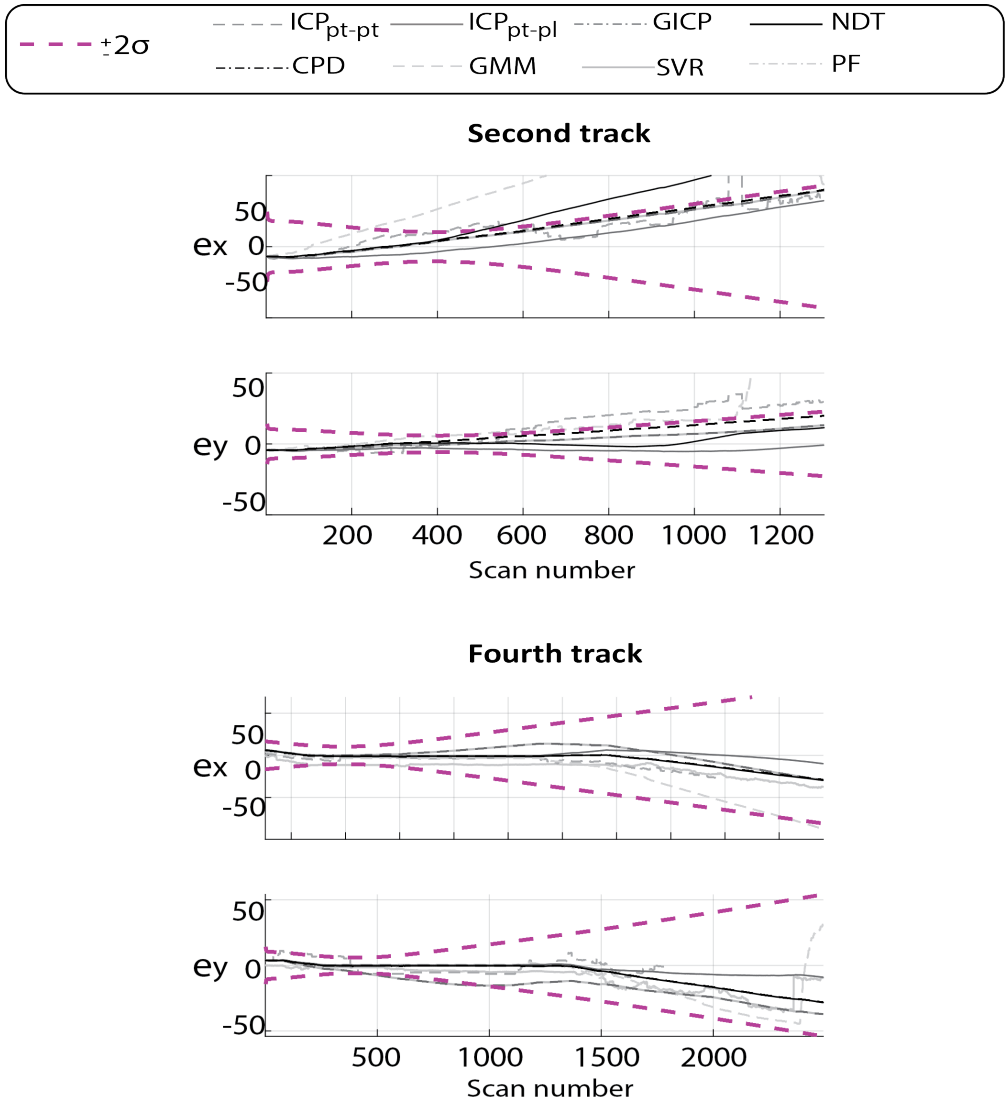
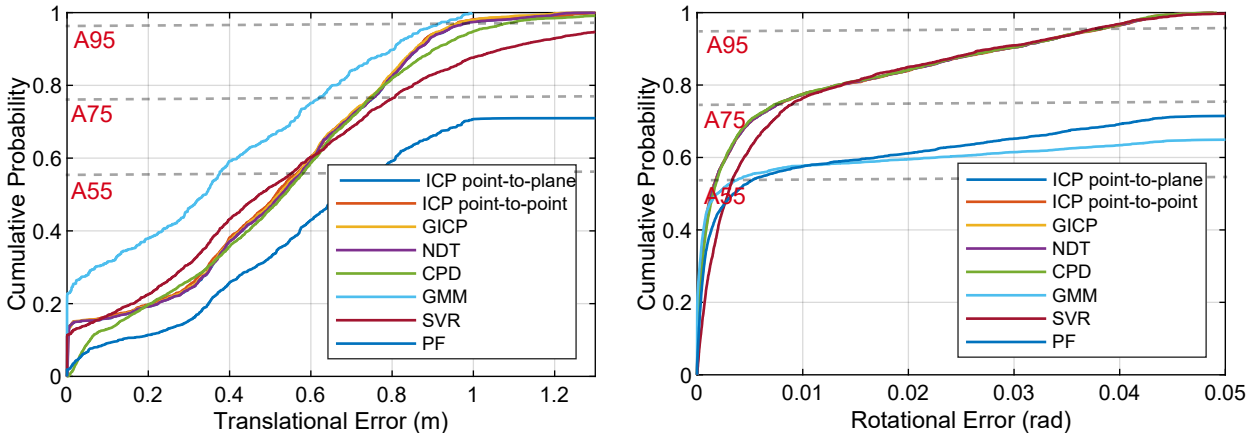


Fig 10: Consistency analysis of the vehicle position and the frame-to-frame registration for a short path experimentation –second and fourth track. The x-axis denotes the scan number. To the left, it is presented the error in the vehicle position estimation e_x and e_y according to x and y coordinates.

error distributions, respectively. An advantage of analyzing the quantiles is a straightforward interpretation of precision and accuracy on the registration procedure.[19] If the difference between the quantiles is small, then the solution is precise. Alternatively, a solution will be more accurate if the error in the quantiles is closer to zero.

Figure 11 and Fig. 12 show the cumulative probability of errors for the urban and agricultural settings, respectively. Such figures present the proportion of outcomes that lie beneath a given error. For the urban

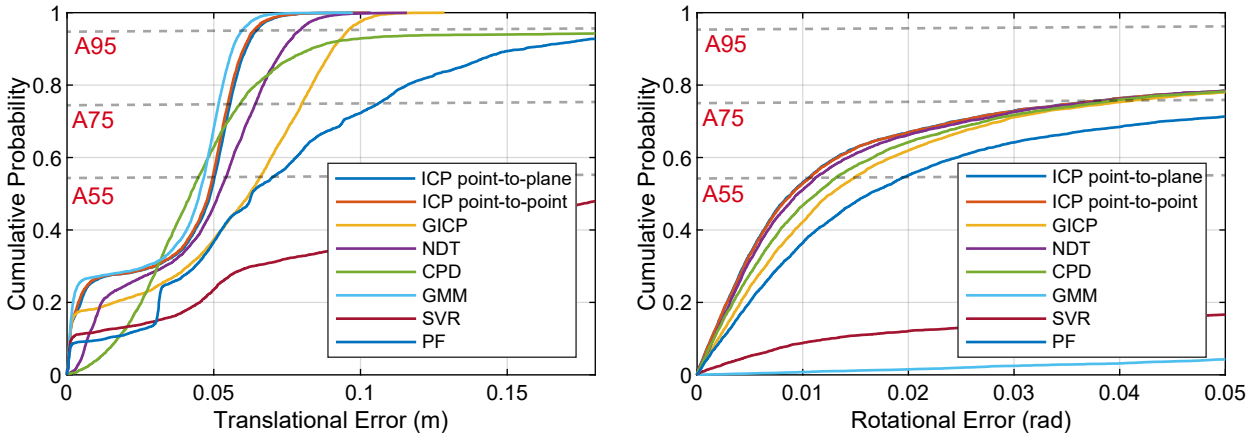
scenario, it is notorious that most of the approaches have similar cumulative distribution for translation and rotation error. Only two approaches have a significantly different response; those are the GMM and the PF. The former seems to outperform the other approaches; however, this outcome is tricky due to as shown in Fig. 7 the GMM outcome do not perceive any movement. The maximum translation estimated in the GMM approach was close to $10e^{-3}$ meters, which tells us that the cumulative probability of the translational error is underestimating the registration error. On the other side, when analyzing the rotational error, it can be seen that the PF and the GMM obtain the worst results, with a high probability of failure.



(a) (b)

Fig 11: Cumulative probability of errors for (a) translation and (b) rotation in the urban dataset. The quantiles of interest A50, A75, A95 are depicted with a gray line.

For the agricultural scenario, it can be seen some differences in the response of the algorithms. It is worth mentioning that the cumulative probability of the translation error showed small values compared to the urban dataset; however, it must be noted that the experiments were acquired with the vehicle at constant and rather low velocity, unlike the urban case. As can be seen in Fig. 12 (a), the SVR and the PF have a higher probability of failure on the translation. The cumulative probability in the rotational error shows a higher probability of errors in all the approaches, having the worst estimation with the GMM and the SVR approaches. It is worth to mention that the variation in the heading of the vehicle between consecutive frames is smaller than the urban environment, as the vehicle traverses the orchards almost in a straight line.



(a) (b)

Fig 12: Cumulative probability of errors for (a) translation and (b) rotation in the Agricultural dataset. The quantiles of interest A50, A75, A95 are depicted with a gray line.

Table 1 summarize the results for the quantiles in translation and rotation for the urban and agricultural settings. The values are depicted in grayscale according to their magnitude; higher values of error are depicted darker. For the translation case, it can be seen that the ICP variations and the NDT are the most precise algorithms for both datasets since the difference between the A95 and A50 is similar and smaller than the other approaches. Meanwhile, the lowest precision in translation was obtained with the SVR and the PF approaches. To analyze the accuracy lets consider the error value in the A50 statistic. The CPD showed the highest accuracy in translation for the urban (0.53 meters) and agricultural (0.04 meters) scenario, and the lowest accuracy was obtained with the PF approach (0.67 meters for the urban and 0.06 meters for the agricultural setting). When analyzing the rotation error, it can be seen that for the urban environment, the GMM and the PF have a lower precision when comparing them to the other approaches. Similarly, in the agricultural case, the GMM and the SVR present higher error values in the rotation. Regarding the accuracy, the ICP variations and the NDT obtain a similar accuracy for the urban and agricultural settings.

Based on the previous results and the calibration procedure developed in Section 4.1, Table 2 summarize the results for precision, accuracy, parameter sensitivity and computational complexity. We established the sensitivity and complexity based on the results of Fig. 5. As previously described, some approaches

Method	Urban						Agricultural					
	Translation (meters)			Rotation (radians)			Translation (meters)			Rotation (randians)		
	A50	A75	A95	A50	A75	A95	A50	A75	A95	A50	A75	A95
ICP_{pt-pt}	0.52	0.73	0.93	14e-4	78e-4	0.03	0.048	0.12	0.06	89e-4	0.04	0.45
ICP_{pt-pl}	0.51	0.71	0.92	13e-4	80e-4	0.03	0.048	0.05	0.06	90e-4	0.04	0.45
$GICP$	0.52	0.72	0.92	14e-4	77e-4	0.03	0.062	0.08	0.09	0.01	0.05	0.43
NDT	0.52	0.73	0.93	14e-4	78e-4	0.03	0.051	0.06	0.07	95e-4	0.05	0.47
CPD	0.53	0.73	1.00	15e-4	77e-4	0.03	0.04	0.05	0.55	0.01	0.04	0.45
GMM	0.34	0.59	0.86	19e-4	0.10	0.12	0.04	0.05	0.05	0.12	0.12	3.08
SVR	0.30	0.78	1.32	29e-4	0.01	0.03	0.19	0.36	0.81	0.51	1.29	2.71
PF	0.67	8.60	30.28	34e-4	1.41	3.11	0.06	0.12	0.42	0.02	0.07	1.08

Table 1: Statistics of the recall-accuracy threshold plots. A50, A75 and A95 denote the probability of 0.5, 0.75 and 0.95 of the errors distribution. The values are depicted in grayscale according to their magnitude.

(ICP variants and NDT) require a few tuning of their parameters to be used, but others may need a more refined process to obtain accurate results. For example, the GMM and the SVR contain several parameters with a significant influence in the estimation, making the calibration procedure not trivial. The computation complexity presented in Table 2 is a key factor in applications with real-time constraints or when the processing power available is limited. For precision and accuracy, we evaluated the A50 and A95 statistics in translation and rotation. First, lets analyze the results for the urban scenario. It is shown that the ICP point-to-point, ICP point-to-plane, GICP, and NDT obtain the highest precision, followed by the CPD and with the lowest precision, the GMM, SVR, and PF. Both the SVR and PF obtain the highest A50 error, thus, the lowest accuracy. The GMM, on the other hand, yields a medium accuracy with a relatively small error in translation, but with the third-highest value in the rotation error. A high accuracy was obtained with the ICP point-to-point, ICP point-to-plane, GICP, NDT, and CPD. Based on the results obtained, we consider that robust (with outlier rejection) ICP point-to-plane should be the first go-to method for any registration problem related to urban environments. Meanwhile, the agriculture environment was a lot more challenging for the presented approaches. As shown in Fig. 8 and in Fig. 12 (b), all the algorithms have a high probability of failure in rotation. These results suggest that the scan matching techniques should not be applied to the raw data due to the highly unstructured environment, and, instead, key points of the orchards should be

used. However, the acquisition of key points using a 3D point cloud representation solely is not an easy task in this type of environment. An alternative, for example, will be to fuse the information of a LiDAR scanner and other exteroceptive sensors such as RGB cameras, obtaining the key points in the RGB domain and projecting them to the 3D point cloud representation. This, of course, will be subjected to the limitations of the camera used.

Method	Precision		Accuracy		Sensitivity	Complexity
	Urban	Agriculture	Urban	Agriculture		
ICP_{pt-pt}	**H**	**M**	**H**	**H**	**L**	**L**
ICP_{pt-pl}	**H**	**M**	**H**	**H**	**L**	**L**
$GICP$	**H**	**L**	**H**	**M**	**L**	**M**
NDT	**H**	**M**	**H**	**H**	**M**	**M**
CPD	**M**	**L**	**H**	**M**	**H**	**H**
GMM	**L**	**L**	**M**	**L**	**H**	**H**
SVR	**L**	**L**	**L**	**L**	**H**	**H**
PF	**L**	**L**	**L**	**L**	**H**	**H**

H:High **M**:Medium **L**:Low

Table 2: The RMSE, translation and rotation error for (a) Ford dataset and (b) Agriculture Dataset

5 Lessons learned

The following are the lessons learned during the experimentation of scan matching techniques in urban and agricultural scenarios.

- The performance of probabilistic approaches relies on the tuning of several parameters with a direct effect on the estimation of the pose of the vehicle. In this work, we used a speed vs. accuracy plot for tuning some of those parameters. However, it is important to consider the trade-off between accuracy and computational time for large point cloud data sets. For example, for the NDT, we prioritized accuracy over computational time, but this will not be a good choice if an embedded system with computational power limitations is used. Other approaches, such as GMM and SVR, need to have a previous adjustment of parameters to avoid inconsistency for each application, resembling an ad-hoc

solution.

- The results obtained from the urban data set show that accurate position estimation can be achieved in relatively long path paths during straight driving. Nevertheless, when the vehicle turns, errors appear in the frame-to-frame registration. Such errors became more evident when the vehicle stopped and objects were moving around it. This outcome remarks one of the main limitations when using scan matching techniques as sole localization strategy.

- In agricultural settings, the major problem was found in the rotation estimation. It was shown that all the scan matching techniques have a high probability of failure in rotation (even for short paths), mainly due to the unstructured shape of these scenarios. Therefore, it seems necessary to consider a pre-processing that could obtain key points on the 3D raw data. This, however, is not a trivial task in unstructured environments.

- The computing time can play a significant role in online applications with real-time constraints or systems with limited processing power. Therefore, the time to convergence could limit the use of some approaches; however, it is not an easy task to get a general evaluation of the time to convergence in all the algorithms because there are several factors that have direct influence, such as the hardware use, the programming language, the amount of parallelism, etc. As recommended in,[19] time should be considered only as a qualitative measure.

- Association among points is a key issue for scan registration. In this context, the experiments show that data association is more accurate when the LiDAR beams collide with continuous and solid surfaces, and it becomes more complex when the LiDAR beams penetrate the objects, as in the agricultural scenario. When the association fails, the optimization for parameters estimation achieves a higher probability of failure, as shown in Figs' consistency tests 10 and 9.

- The velocity of the mobile platform directly impacts on the overlap between two consecutive scans.

When the travel velocity increases, the overlap decreases. Future research should be devoted to better understand the overlapping region of influence and its effect on the accuracy of the scan matching approaches.

- Future work should include the assessment of other sensors for scan matching purposes, thus enhancing the information managed by the approach. Additionally, the influence of 3D sensors performance (accuracy, precision, resolution) and their effects on the scan matching accuracy should be further investigated.

- Finally, as shown in Figs. 7 and 8, the scan matching techniques as localization strategies in autonomous vehicles should be used as a complementary technique to GNSS antennas (or another absolute localization system), since it is susceptible to falling into inconsistency, thus jeopardizing the autonomy and safety of the vehicle (and its possible passengers in urban applications).

6 Conclusions

In this work, we focused on the localization of autonomous vehicles based on the registration of 3D point clouds acquired at different times from a LiDAR sensor. Eight different scan matching algorithms that represent the current state-of-the-art in the field were investigated, namely, ICP, its variations to point-to-point and to point-to-plane, GICP, NDT, GMM, CPD, SVR, and PF. The choice for one of these algorithms generally depends on several important characteristics such as accuracy, computational complexity, and convergence rate, each of which depends, in turn, on the application of interest. To the best of our knowledge, a general discussion of each of the above methods is not available in the literature. The algorithms were tested using 3D data of two types of outdoor environments: urban and agricultural. Results showed that the performance of most registration algorithms heavily depends on the data used, and thus on the environment itself. The agricultural setting proved to be more challenging due to natural objects with less structured features than urban scenarios. The results presented herein are intended to encourage researchers and developers to build

improvements on the identified limitations by developing new scan matching systems based on more robust algorithms and more accurate 3D sensors.

Acknowledgement(s)

This project has been supported by the National Agency of Research and Development (ANID, ex-Conicyt) under Fondecyt grant 1201319, Basal grant FB0008, DGIIP-UTFSM Chile, National Agency for Research and Development (ANID)/PCHA/Doctorado Nacional/2020-21200700, Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya (grant 2017 SGR 646), the Spanish Ministry of Science, Innovation and Universities (project RTI2018- 094222-B-I00) for partially funding this research. The Spanish Ministry of Education is thanked for Mr. J. Gené's pre-doctoral fellowships (FPU15/03355). We would also like to thank Nufri (especially Santiago Salamero and Oriol Morreres) for their support during data acquisition

Appendix. Source codes

The dataset used in this work can be found in:

- **Ford Dataset (Urban)** http://robots.engin.umich.edu/SoftwareData/Ford
- **Lleida Dataset (Agriculture)** http://www.grap.udl.cat/en/publications/datasets.html

Following, the repositories of the source codes used in this tutorial. Please refer to Section II for the mathematical background and to Section III for the design parameters adopted in each algorithm.

- **ICP** https://www.mathworks.com/help/vision/ref/pcregistericp.html
- **G-ICP** https://pointclouds.org/
- **NDT** https://pointclouds.org/

- **CPD** https://github.com/gadomski/cpd
- **SVR** https://github.com/neka-nat/probreg
- **GMM** https://github.com/bing-jian/gmmreg
- **Particle Filter** https://github.com/DrGabor/Vanilla_PF_ICP

References

1 F. Dong, Y. Chen, J. Liu, *et al.*, "Mobile Entity Localization and Tracking in GPS-less Environments," in *MELT '09. Proceedings of the 2nd international conference on Mobile entity localization and tracking in GPS-less environments*, (May), 79–90–90 (2009).

2 J. G. Mooney and E. N. Johnson, "A Comparison of Automatic Nap-of-the-earth Guidance Strategies for Helicopters," *Journal of Field Robotics* **28**(5), 1–17 (2014).

3 J. Kaivosoja and R. Linkolehto, "GNSS error simulator for farm machinery navigation development," *Computers and Electronics in Agriculture* **119**, 166–177 (2015).

4 R. Ren, H. Fu, and M. Wu, "Large-scale outdoor slam based on 2d lidar," *Electronics (Switzerland)* **8**(6) (2019).

5 W. S. Grant, R. C. Voorhies, and L. Itti, "Efficient Velodyne SLAM with point and plane features," *Autonomous Robots* **43**(5), 1207–1224 (2019).

6 S. Thrun, "Probabilistic robotics," *Communications of the ACM* **45**(3), 52–57 (2002).

7 H. Choset, K. Lynch, S. Hutchinson, *et al.*, *Principles of Robot Motion: Theory, Algorithms and Implementation* (2005).

8 F. Hoflinger, J. Muller, R. Zhang, *et al.*, "A wireless micro inertial measurement unit (IMU)," *IEEE Transactions on Instrumentation and Measurement* **62**(9), 2583–2595 (2013).

9 F. Aghili and C. Y. Su, "Robust relative navigation by integration of ICP and adaptive Kalman filter using laser scanner and IMU," *IEEE/ASME Transactions on Mechatronics* **21**(4), 2015–2026 (2016).

10 F. B. Malavazi, R. Guyonneau, J. B. Fasquel, *et al.*, "LiDAR-only based navigation algorithm for an autonomous agricultural robot," *Computers and Electronics in Agriculture* **154**(February), 71–79 (2018).

11 M. Varela-González, H. González-Jorge, B. Riveiro, *et al.*, "Automatic filtering of vehicles from mobile LiDAR datasets," *Measurement: Journal of the International Measurement Confederation* **53**, 215–223 (2014).

12 M. Bueno, H. González-Jorge, J. Martínez-Sánchez, *et al.*, "Evaluation of point cloud registration using Monte Carlo method," *Measurement: Journal of the International Measurement Confederation* **92**, 264–270 (2016).

13 F. Auat Cheein, M. Torres-Torriti, and J. R. Rosell-Polo, "Usability analysis of scan matching techniques for localization of field machinery in avocado groves," *Computers and Electronics in Agriculture* **162**(May), 941–950 (2019).

14 T. Viking, M. Pathfinder, and M. Explo, "Real Time Egomotion of a Nonholonomic Vehicle using LIDAR Measurements," *Journal of Field Robotics* **24**(4), 273–274 (2007).

15 L. Montesano, J. Minguez, and L. Montano, "Probabilistic scan matching for motion estimation in unstructured environments," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 1445–1450 (2005).

16 D. Borrmann, J. Elseberg, K. Lingemann, *et al.*, "Globally consistent 3D mapping with scan matching," *Robotics and Autonomous Systems* **56**(2), 130–142 (2008).

17 P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**(2), 239–256 (1992).

18 A. Milella, G. Reina, and R. Siegwart, "Computer vision methods for improved mobile robot state estimation in challenging terrains," *Journal of Multimedia* **1**(7), 49–61 (2006).

19 F. Pomerleau, F. Colas, R. Siegwart, *et al.*, "Comparing ICP variants on real-world data sets: Open-source library and experimental protocol," *Autonomous Robots* **34**(3), 133–148 (2013).

20 F. A. Donoso, K. J. Austin, and P. R. McAree, "How do ICP variants perform when used for scan matching terrain point clouds?," *Robotics and Autonomous Systems* **87**, 147–161 (2017).

21 Y. Tian, X. Liu, L. Li, *et al.*, "Intensity-assisted ICP for fast registration of 2D-LIDAR," *Sensors (Switzerland)* **19**(9) (2019).

22 G. Huo, L. Zhao, K. Wang, *et al.*, "Polar Metric-Weighted Norm-Based scan matching for robot pose estimation," *Discrete Dynamics in Nature and Society* **2016** (2016).

23 F. Lu and E. Milios, "Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans," *Journal of Intelligent and Robotic Systems: Theory and Applications* **18**(3), 249–275 (1997).

24 A. V. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," *Robotics: Science and Systems* **2**(4), 435 (2009).

25 M. Magnusson, A. Nüchter, C. Lörken, *et al.*, "Evaluation of 3D registration reliability and speed-A comparison of ICP and NDT," in *Proceedings - IEEE International Conference on Robotics and Automation*, 3907–3912 (2009).

26 A. Zaganidis, M. Magnusson, T. Duckett, *et al.*, "Semantic-assisted 3D normal distributions transform for scan registration in environments with limited structure," in *IEEE International Conference on Intelligent Robots and Systems*, **2017-Septe**, 4064–4069 (2017).

27 A. Zaganidis, L. Sun, T. Duckett, *et al.*, "Integrating Deep Semantic Segmentation into 3-D Point Cloud Registration," *IEEE Robotics and Automation Letters* **3**(4), 2942–2949 (2018).

28 C. R. Qi, H. Su, K. Mo, *et al.*, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, **2017-Janua**, 77–85 (2017).

29 B. Eckart, K. Kim, and J. Kautz, "Fast and Accurate Point Cloud Registration using Trees of Gaussian Mixtures," (2018).

30 X. Huang, J. Zhang, Q. Wu, *et al.*, "A Coarse-to-Fine Algorithm for Matching and Registration in 3D Cross-Source Point Clouds," *IEEE Transactions on Circuits and Systems for Video Technology* **28**(10), 2965–2977 (2018).

31 D. Campbell and L. Petersson, "An adaptive data representation for robust point-set registration and merging," in *Proceedings of the IEEE International Conference on Computer Vision*, **2015 Inter**, 4292–4300 (2015).

32 S. Billings and R. Taylor, "Generalized iterative most likely oriented-point (G-IMLOP) registration," *International Journal of Computer Assisted Radiology and Surgery* **10**(8), 1213–1226 (2015).

33 R. A. Srivatsan, G. T. Rosen, D. F. N. Mohamed, *et al.*, "Estimating SE(3) elements using a dual quaternion based linear Kalman filter," in *Robotics: Science and Systems*, **12**(3) (2016).

34 W. Tao, X. Hua, K. Yu, *et al.*, "An improved point-to-plane registration method for terrestrial laser scanning data," *IEEE Access* **6**, 48062–48073 (2018).

35 J. Serafin and G. Grisetti, "NICP: Dense normal based point cloud registration," in *IEEE International Conference on Intelligent Robots and Systems*, **2015-Decem**, 742–749 (2015).

36 M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," (2007).

37 L. Maier-Hein, A. M. Franz, T. R. Dos Santos, *et al.*, "Convergent iterative closest-point algorithm to accomodate anisotropic and inhomogenous localization error," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(8), 1520–1532 (2012).

38 S. Billings and R. Taylor, "Iterative most likely oriented point registration," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, **8673 LNCS**(PART 1), 178–185 (2014).

39 S. Granger and X. Pennec, "Multi-scale EM-ICP: A fast and robust approach for surface registration," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, **2353**, 418–432 (2002).

40 G. Agamennoni, S. Fontana, R. Y. Siegwart, *et al.*, "Point Clouds Registration with Probabilistic Data Association," in *IEEE International Conference on Intelligent Robots and Systems*, **2016-Novem**, 4092–4098 (2016).

41 M. Magnusson, *The Three-Dimensional Normal-Distributions Transform — an Efficient Representation for Registration, Surface Analysis, and Loop Detection*. PhD thesis, Örebro University (2009).

42 M. Lu, J. Zhao, Y. Guo, *et al.*, "Accelerated Coherent Point Drift for Automatic Three-Dimensional Point Cloud Registration," *IEEE Geoscience and Remote Sensing Letters* **13**(2), 162–166 (2016).

43 B. Jian and B. C. Vemuri, "Robust point set registration using Gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(8), 1633–1645 (2011).

44 Q. Li, R. Xiong, and T. Vidal-Calleja, "A GMM based uncertainty model for point clouds registration," *Robotics and Autonomous Systems* **91**, 349–362 (2017).

45 R. Sandhu, S. Dambreville, and A. Tannenbaum, "Point set registration via particle filtering and stochastic dynamics," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(8), 1459–1473 (2010).

46 M. S. Arulampalam, S. Maskell, N. Gordon, *et al.*, "A tutorial on particle filters for online nonlinear/nongaussian bayesian tracking," in *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*, **50**(2), 723–737 (2002).

47 G. Pandey, J. R. McBride, and R. M. Eustice, "Ford Campus vision and lidar data set," *International Journal of Robotics Research* **30**(13), 1543–1552 (2011).

48 U. Meier, *Growth stages of mono- and dicotyledonous plants*, vol. 12, Blackwell Wissenschafts-Verlag (2001).

49 T. Palleja, M. Tresanchez, M. Teixido, *et al.*, "Sensitivity of tree volume measurement to trajectory errors from a terrestrial LIDAR scanner," *Agricultural and Forest Meteorology* **150**(11), 1420–1427 (2010).

50 J. Gené-Mola, R. Sanz-Cortiella, J. R. Rosell-Polo, *et al.*, "Fruit detection and 3D location using instance segmentation neural networks and structure-from-motion photogrammetry," *Computers and Electronics in Agriculture* **169**(December 2019), 105165 (2020).

51 F. Pomerleau, S. Magnenat, F. Colas, *et al.*, "Tracking a Depth Camera: Parameter Exploration for Fast ICP," (2015).

52 M. Magnusson and T. Duckett, "A Comparison of 3D Registration Algorithms for Autonomous Underground Mining Vehicles," in *Proceedings of the European Conference on Mobile Robotics (ECMR 2005)*, 86–91 (2005).

53 P. Babin, P. Giguere, and F. Pomerleau, "Analysis of Robust Functions for Registration Algorithms," (October), 1451–1457 (2019).

54 M. Lamine Tazir, T. Gokhool, P. Checchin, *et al.*, "CICP: Cluster Iterative Closest Point for sparse–dense point cloud registration," *Robotics and Autonomous Systems* **108**, 66–86 (2018).

55 B. Eckart, K. Kim, and J. Kautz, "HGMR: Hierarchical gaussian mixtures for adaptive 3D registration," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, **11219 LNCS**, 730–746 (2018).

56 X. Gu, X. Wang, and Y. Guo, "A Review of Research on Point Cloud Registration Methods," *IOP Conference Series: Materials Science and Engineering* **782**(2) (2020).

57 F. Pomerleau, F. Colas, and R. Siegwart, "A Review of Point Cloud Registration Algorithms for Mobile Robotics," *Foundations and Trends in Robotics* **4**(1), 1–104 (2015).

58 C. Ulaş and H. Temeltaş, "3D multi-layered normal distribution transform for fast and long range scan matching," *Journal of Intelligent and Robotic Systems: Theory and Applications* **71**(1), 85–108 (2013).

59 T. Zhang, K. Wu, J. Song, *et al.*, "Convergence and Consistency Analysis for a 3-D Invariant-EKF SLAM," *IEEE Robotics and Automation Letters* **2**(2), 733–740 (2017).

Darío Guevara received the B.S. degree in electronics and control engineering from Escuela Politécnica Nacional, Quito, Ecuador, in 2016. Since 2017, he is currently pursuing the Ph.D. degree from the Department of Electronic Engineering, Universidad Técnica Federico Santa María, Valparaíso, Chile. His research interest include sensing technologies, sensor fusion, machine learning and mobile robotics.

Jordi Gené-Mola received the B.S. degree in Mechanical Engineering and the M.Sc. degree in Industrial Engineering from the Universitat de Lleida, in 2013 and 2015 respectively, the M.Sc. degree in Computer Vision from the Universitat Autònoma de Barcelona, in 2018, and the Ph.D. degree from the Universitat de Lleida, in 2020. He is currently working as postdoctoral researcher in the Research Group in Precision Agriculture (GRAP), Department of Agricultural and Forest Engineering, University of Lleida. His research interests include the use of sensors in agriculture, electronic characterization of vegetation and computer vision in agriculture.

Eduard Gregorio López received the B.S. degree in Mechanical Engineering from the Universitat de Lleida, Spain, in 2003, the M.Sc. degree in Industrial Engineering from the Universitat Politécnica de Catalunya, Spain, in 2005, and the Ph.D. degree from the Universitat de Lleida, in 2012. Since 2009, he has been with the Department of Agricultural and Forest Engineering, Universitat de Lleida, where he is currently an Associate Professor and a member of the Research Group in Precision Agriculture (GRAP). His research interests include the use of sensors in agriculture, light detection and ranging, computer vision in agriculture and electrical engineering.

Miguel Torres-Torriti received the B.Sc. and M.Sc. degrees in electrical engineering from the Pontificia Universidad Catolica de Chile (PUC-Chile), in 1996 and 1998, respectively, and the Ph.D. degree from

McGill University, Montreal, Canada, in 2003. Between 2004 and 2005 he was Senior Applications Engineer with General Electric Chile in the implementation of the multivariable process controllers for the ENAP Bio and Magallanes refineries. In 2005 he joined the Department of Electrical Engineering of the School of Engineering of PUC Chile, where he is currently Associate Professor, Director of the undergraduate program in Autonomous and Robotic Systems, and Associate Dean of Social Responsibility. His fields of experience comprise systems modelling and control, estimation, robot dynamics, sensors and perception. His current research projects include the development of navigation strategies for mining and agricultural machines considering interactions with the environment, data fusion of satellite imagery and wireless sensor networks for early warning of flash-floods, and the development of assistance devices for persons with motor disabilities.

Giulio Reina received the Laurea degree and the Research Doctorate degree from the Politecnico of Bari, Italy in 2000 and 2004 respectively, both in Mechanical Engineering. He was a Visiting Scholar at the University of Michigan's Mobile Robotics Laboratory, and a JSPS and Endeavour Research fellow, respectively, at the Space Robotics Laboratory of Tohoku University, Sendai, Japan, and Australian Centre for Field Robotics (ACFR) of the University of Sydney, Australia. Currently, he serves as an Associate Professor at the Department of Mechanics, Mathematics and Management, Politecnico di Bari, Italy. His research interests include mobile robotics for planetary exploration, vehicle dynamic modelling and estimation, and advanced driving assistance systems (ADAS) for automotive and agriculture industry towards self-driving vehicles.

Fernando Auat Cheein received the B.S. degree in Electronics Engineering from Universidad Nacional de Tucuman, Argentina, in 2002 and the M.S. and Ph.D. degree in Control Systems from Universidad Nacional de San Juan, Argentina, in 2005 and 2009 respectively. Since 2011, he has been an Assistant Professor with the Department of Electronic Engineering, Universidad Técnica Federico Santa María, Valparaíso, Chile.

He is founder of GRAI (Autonomous and Industrial Robotics Research Group) and part of the board of the Advanced Center for Electrical and Electronic Engineering in Chile.

List of Figures

7 Paths reconstruction. Ground truth path shown in solid magenta line and estimated path shown in solid dark line. In gray, it is shown the 2D projection of the reconstructed map.

8 Results from the agricultural environment. On top, the paths followed by the vehicle; S_p denotes the starting point of each path. There are two short path trials, starting at S_{p2} and S_{p4}. The long path experimentation starts at S_{p5}. The medium path trials are labelled in red and blue, respectively, while the short paths are depicted in cyan; the long path is shown in purple. In yellow the rest of the point cloud obtained during trials. On the bottom, it is shown the estimated path according to each registration technique.

9 Consistency analysis of the vehicle position and the frame-to-frame registration for the medium path experimentation –first and third tracks–, and for the long path experimentation –fifth track. The left figure shows the positioning errors e_x and e_y according to x and y coordinates.

10 Consistency analysis of the vehicle position and the frame-to-frame registration for a short path experimentation –second and fourth track. The x-axis denotes the scan number. To the left, it is presented the error in the vehicle position estimation e_x and e_y according to x and y coordinates.

11 Cumulative probability of errors for (a) translation and (b) rotation in the urban dataset. The quantiles of interest A50, A75, A95 are depicted with a gray line.

12 Cumulative probability of errors for (a) translation and (b) rotation in the Agricultural dataset. The quantiles of interest A50, A75, A95 are depicted with a gray line.

List of Tables